(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 674 440 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **95103794.4**

(51) Int. Cl.[6]: **H04N 7/167**

(22) Date of filing: **16.03.95**

(30) Priority: **21.03.94 FI 941316**

(43) Date of publication of application:
27.09.95 Bulletin 95/39

(84) Designated Contracting States:
DE FR GB IT

(71) Applicant: **NOKIA TECHNOLOGY GmbH**
Östliche Karl-Friedrich-Strasse 132
D-75175 Pforzheim (DE)

(72) Inventor: **Kangas, Mauri**
Sporentie 21
SF-21530 Paimio (FI)

(54) A process for encryption and decryption of a bit stream containing digital information.

(57) According to the invention digital video, audio and data information can be encrypted according to the MPEG-2 standard either by encrypting the PES packets with an accuracy of a bit or a byte, or by encrypting the transport stream packet with block encryption. Both encryptions can be used in combination. The decryption device in a receiver identifies from the transport stream packet header on which level the encryption is made at the transmitting end, and controls the decryption in accordance with that identification.

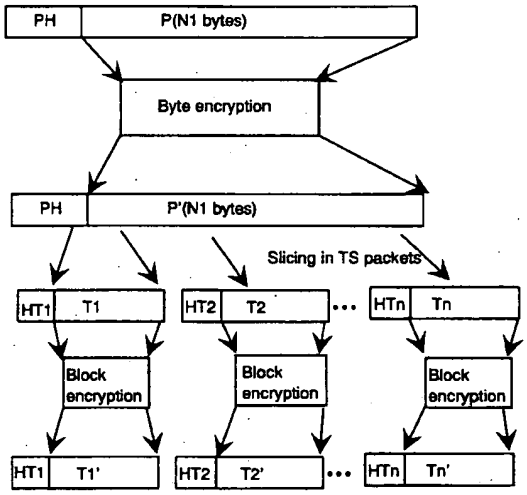

FIG. 10

EP 0 674 440 A2

The invention relates to a process for encryption and decryption of digital video, audio and data signals transmitted on a transmission path. In the process the video, audio and data information contained in the digital bit stream is first encrypted in an encryption device at the transmitting end, and then decrypted in the receiving device.

For the transmission of digital video, audio and data signals we can use transmission via air through a land based antenna, a satellite connection, a cable TV network, the telephone/telecommunications network, or an optical cable network, through which the information is transmitted to a plurality of receivers. Said information is often intended to be freely used by all receivers, but on the other hand also such methods are needed, by which it is possible to control which receiver/receivers actually can receive the information. Thus, in the transmission of an analog video signal in a pay TV system and in corresponding systems, the subscriber selects a program which she/he desires, and pays for those programs which are transmitted as encrypted signals. In the analog systems the encryption can be based on shuffling the order of the picture lines by blocks, which are smaller than the picture area. The subscriber has obtained an encryption key against payment. In advanced pay TV systems it is even possible to buy the rights to view an encrypted program a few moments before the transmission starts of even during the transmission. Then the subscriber in advance has loaded electronic money in a smart card for instance, and if there is a balance on the card, then the selected encrypted program is decrypted so that it can be viewed. It has also been proposed to use an arrangement in which a certain code stating the price of the program is attached to the transmitted program signal. The receiver compares the payment with the loaded money, and the program is decrypted when there is a sufficient balance.

The present encryption methods used mainly in pay TV applications are based on the encryption of an analog video signal. These methods cannot be directly applied to future digital systems, and they do not utilize the characteristics of digital transmission. The object of pay TV operators is an encryption process which in the receiver of a paying viewer produces picture and sound of good quality, which in other words must not corrupt the transmitted program. From the operators' point of view it is also favorable if it is possible to view the encrypted transmission to some degree without decryption. In this case the quality of the picture and the sound must be sufficiently good, so that the viewer can have an idea of this operator's programs, but on the other hand it must have a sufficiently bad quality, so that the program can not be viewed with pleasure. To an operator an encrypted transmission provides then a means to advertise the operator's programs to those viewers who are in the same network but who have not paid to receive the program in question.

A plurality of encryption methods are available when the transmitted information is in digital form and also contains something else than pure video information, as is assumed below. When we wish to control which receiver or receivers actually can receive the transmitted digital information, then there are basically two lines of action:

a) the digital information is transmitted according to a predetermined plan, whereby each receiver can express his/her desire to receive that information prior to the transmission or during it;

b) the digital information is transmitted only when any receiver expresses his/her desire to receive this information.

In the latter case also other than that individual who ordered can be allowed to receive this information, whereby these other receivers can already earlier have obtained the authorization to receive this transmission, or they can order it during the transmission. As it is desirable to prevent unauthorized receiving of the signal in such digital signal transmission, the transmitted bit stream is transformed into a form for which the receivers have no possibility to disentangle the data contents if they do not have the keys required for data decryption.

Generally there are many methods available for the encryption of a digital bit stream, and in order to make impossible an unauthorized attempt to decrypt the data when the system has been commissioned, it is desirable to have an encryption system which is as complicated and as secure as possible. In order to maximize the security it is advisable to use a so called block encryption algorithm whenever it is technically feasible. The algorithm will divide into blocks the data stream to be encrypted, which could comprise only a section of the total data stream. The block size could be e.g. 8 bytes, and the encryption is made in one operation for the whole block. In such cases where it is cumbersome to divide the data information into desired blocks, it is however possible to use a so called PRBS generator (Pseudo-Random Bit Sequence), whereby the data section to be encrypted can have any number of bytes, even a precision of one bit could be achieved when needed. If we want to guarantee the algorithm's security against unauthorized decryption attempts it is advisable to combine two of the above mentioned algorithms. Regarding the block encryption algorithm the most straightforward way is to keep within the limits imposed by the block size.

The transmission of an audio signal is an essential part of the transmission of a digital signal,

but also so called control data has to be transmitted when systems are realized. In the future it is also necessary to transmit so called data information in order to have more versatile services, whereby the data information can contain almost any kind of information from the system's point of view. All these information sections must be transmitted in encrypted form, at least partly, in order to secure that only authorized receivers can receive the information.

Future digital television systems enable simultaneous transmission of several programs in one transmission channel. Then the transmitted signal is in packet form, and the transmission channel sequentially transports packets containing audio and video information of these different programs. One video packet usually contains information of several picture blocks, whereby one picture block can comprise e.g. 8x8 pixels, or the video data packet can contain picture information of so called macro blocks, comprising 16x16 pixels. It is also possible to transmit data packets which are attached to the programs. MPEG-2 (Motion Picture Experts Group) is a generic standard of high quality video compression methods, with which a television picture can be transmitted in fewer bits than when the television picture is digitized directly into bits.

Several compression standards have been developed in order to transmit video, audio and data signals, the above mentioned MPEG-2 (Motion Picture Experts Group) being one of these. This standard was developed in a joint work group of ISO (International Standards Organization) and IEC (International Electrotechnical Commission). Several MEPG standards have been developed, and in future the transmission of the above mentioned information will be realized in many different applications, according to the specifications defined by these standards. Concerning the standards we refer to the MPEG standard ISO/IEC 13818 known by persons skilled in the art. According to this standard the encoded video, audio and data information is packed into so called PES packets (Packetized Elementary Stream). The packet will contain a header, and data section and the whole packet may have varying lengths.

Figure 1 shows the structure of a PES packet. The packet header comprises the packet start code prefix, the stream ID, an indication of the packet length, an optional header, and a plurality of stuffing bytes. Then there is an information section containing the data bytes of the packet, and as was mentioned above, the information section may comprise a block of the program's encoded audio, video or data signals, but so that one packet contains a signal of one type only. The packet can have a length of several kilobytes.

The MPEG standard defines two bit streams of different types: 1) the Program Stream, and 2) the Transport Stream. The program stream contains the encoded video and audio signal in the form of PES packets (Packetized Elementary Stream) referred to above, each of these packets always containing a bit stream block of a certain size, in other words so that the video, audio and data signal of the program source is encoded separately and sliced into blocks of a certain length, each block being placed in the information section of the PES packet. The length of a block and thus the length of the PES packet may vary. Thus the program's video signal comprises sequential video-PES packets the audio signal comprises sequential audio-PES packets, and so on. For instance all information of a motion picture may be stored as a program stream.

Figures 2a and 2b show how the PES packets of figure 1 are placed as packets in the program stream. The program stream, figure 2a, comprises sequential packets having a packet header and an information section PACK. Figure 2b showing the structure of one packet in the program stream, the so called packet level, illustrates how a program stream packet PACK contains several PES packets, which are numbered #1, #2, ..., #n, and which may contain picture, sound, data etc. relating to the program. For the sake of clarity we can consider that the packet sequence of figure 2a represents e.g. one motion picture the PES packetized audio and video signals of which are placed in the information sections of the program stream packets. The program stream ends in the "program end code".

When a program shall be transmitted on a transmission link, then a so called transport stream is formed, which is intended for the transmission of audio and video signals on any transmission link, such as TV broadcasts, satellite, cable TV, telephone/telecommunications cables, optical cables, etc. If the program source is a recording in the form of a program stream, e.g. a motion picture recorded on a CD disk, the program stream is first demultiplexed into separate audio, video and data PES packets. On the other hand, if the program source provides audio, video and data signals, then these are decoded and formed into PES packets. Whatever the source may be, the PES packets provided by the source are placed into the transport stream.

Figures 3a and 3b show the structure of the transport stream. The transport stream is such that it comprises transport stream packets having a fixed length of e.g. 188 bytes figure 3a. A packet comprises a header of varying length, and a data section containing the useful information or payload. Figure 3b shows the structure of one trans-

port stream packet. The packet header comprises 9 fields, which will not be described in more detail here. Here we must observe the last field or the adaptation field, which plays an important role, as is described later. The PES packet bytes are included in the packet payload.

Let us consider how the PES packets of figure 1 are placed in the transport stream packet of figure 3b. After the transport stream packet header there is first the PES packet header, which indicates the start of a new PES packet. Then the payload section is filled with PES packet bytes starting at the beginning of the packet. Then the operation moves to fill the next payload section with PES packet bytes. This is continued until the last PES packet bytes are placed in the last transport stream packet. If there are less bytes than the place reserved for the actual payload, then the header adaptation part is filled with as many stuffing bits as are necessary to obtain the standard length of the transport stream packet. After the header of the next transport stream packet there is the header of the next PES packet, and after this the payload section where the bytes of this next PES packet are placed. Then the operation continues to the next transport stream packet and filling its payload section, an so on. Here it must be observed that the border of two PES packets divided into the transport stream is never inside a transport stream packet, but a new PES packet always starts after the header of the transport stream packet, and the packet is ended so that its last bytes at the same time form the last bytes of a transport stream packet.

In must be noted here that the longest PES packet could be several kilobytes, and thus several times longer than a transport stream packet of the fixed length (188 bytes). Thus the transport stream packets, figure 3, comprise PES packet sections of varying lengths. Because the PES packets contained in a program stream are divided rather straightforwardly into the packets of the frequency stream, one PES packet contained in the program stream is divided into very many transport stream packets.

Depending on the application the digital bit stream may exist either as a program stream or a transport stream, or also in a form where the PES packets contained in the program stream are independent entities. Video and audio signals have their own PES packets, whereby a program stream packet contains a varying number of packets in varying locations within the program stream. Because there may arise situations in which, on the other hand, the digital information should be processed as a program stream, as PES packets, or as a transport stream, and in which on the other hand it is desirable to encrypt the bit streams in the form they happen to have in the respective application, then it is reasonable to be able to encrypt the bit streams on all respective levels. Then the encryption should have at least two alternatives:

1) encryption on the transport stream level, and
2) encryption on the PES level.

Because the transport stream level is the more common means to transmit and process data, it is reasonable that the encryption is as secure as possible at least on this level. Because the transport stream is divided into packets of fixed lengths, as was stated above, it is logical to have the data stream encryption on the transport stream level made as an operation which concerns single packets.

Because the length of the section to be encrypted contained in the packets in the transport stream may vary, it is not possible to have a block encryption to encrypt directly the whole section to be encrypted, as its length very seldom comprises a multiple of the encryption block. Therefore prior art systems use as the encryption algorithm for the transport stream a combination of block encryption and a PRBS generator, with which it is possible to obtain a sufficient encryption security. By using a combination of these two algorithms it is possible to overcome the block size limitation imposed by the block encryption, because an encryption multiple remainder can be encrypted only by the PRBS generator which can provide an accuracy of one byte regarding the length of the section to be encrypted.

In the encryption on the program stream level or on the PES level the prior art systems will process packets of varying lengths which comprise sections to be encrypted of correspondingly varying lengths. When the program stream or the individual PES packets are directed into the transport stream, then the PES packets have to be sliced pieces suitable for the transport stream. This presents at least the following problems: If the PES packet was encrypted on the program stream level or on the PES level, then during the encryption process it is not possible to know, expect in special cases, the sizes of the blocks into which the PES packet will be sliced. If block encryption is used for the entire PES packet, then the slicing operation most often will lead to a situation where the slicing will happen at a place within the encryption block, so that the encryption block is separated in two transport stream blocks. This in turn will lead to a complicated structure in the decryption device, because sequential packets or at least packet sections must be stored there in the internal memory in order to finish the decryption process. On the other hand it is desirable to use this decryption device to decrypt several sequential data streams,

and on the other hand the sequential packets of the same data stream will not be sequential in the transport stream. This will in turn lead to a situation in which the sequential packets or sections of these packets must be temporarily stored in memory in order to decrypt all packets in the transport stream. If the decryption device is realized according to this principle it will be unnecessarily complicated.

In order to avoid the above mentioned problems the following requirements must be satisfied when the data stream encryption is realized according to the MPEG standard:

- It must be possible to encrypt the data stream on the program stream level and the transport stream level as well as on the PES packet level.
- The encryption must by as secure as possible.
- The decryption must be realized as simply as possible within the limits set by the basic algorithms.

The object of the process to use the encryption algorithms according to the patent claims is to satisfy all the requirements presented above. The process to use the encryption algorithms does not actually take a stand on the actual algorithms, but of course we select algorithms which are as secure as possible and which shuffle the bit stream as effectively as possible.

According to the basic idea of the invention we use two different encryption algorithms simultaneously, but on different levels according to the MPEG standard. One algorithm performs block encryption on one level, whereby the whole section to be encrypted must be a multiple of the block length. Accordingly the second algorithm performs the encryption on a second level with an accuracy of one byte, even one bit, using the PRBS algorithm. Below in the text we call this latter encryption method byte encryption. When necessary we use the abbreviation TS for the transport stream. In addition to the basic method, which comprises simultaneous use of block encryption and byte encryption, the invention comprises several embodiments to perform the encryption.

According to the first embodiment the encryption is made on the PES level by using byte encryption, and on the TS level there is no encryption. According to the second embodiment there is no encryption on the PES level, but on the TS level there is performed either a block encryption with an accuracy of a block or a partly double encryption, which also comprises bytes outside the encryption block multiple.

According to the third embodiment byte encryption is made on the PES level, and on the TS level there is performed block encryption with an accuracy of the block size or a partly double encryption comprising also the bytes outside of the block size multiple.

The different embodiments of the invention are described below with reference to the enclosed schematic figures, in which:

figure 1 shows a PES packet according to the MPEG-2 standard;
figure 2 shows the principle of the program stream;
figures 3a and 3b show the principle of the transport stream;
figure 4 shows the encryption according to the first embodiment;
figure 5 shows the encryption according to the second embodiment;
figure 6 shows a chained block encryption of the TS packet;
figure 7 shows the decryption of the chained block encryption of the TS packet;
figure 8 shows a direct block encryption of the TS packet;
figure 9 shows the decryption of the direct block encryption of the TS packet;
figure 10 shows the encryption according to the third embodiment;
figure 11 shows a combined byte encryption and block encryption;
figure 12 shows the decryption of the encryption according to figure 11; and
figure 13 illustrates how the block encryption and the byte encryption is made in the same process.

Figure 4 shows how the encryption is made according to the first embodiment. The encryption is made on the PES level using byte encryption, i.e. a PRBS generator is used. The PES packet to be encrypted comprises a header PH, after which there is the information section P(N1) comprising N1 bytes. Here it should be observed that both the header length and the information section length, and thus the total packet length can vary within certain limits. The byte encryption is made only on the information section, and there is no reason to encrypt the header, because the decryption is made easier when the header is not encrypted. After the byte encryption the original PES packet has transformed into a packet comprising the original header PH and the information section P' (N1) which was transformed by the encryption algorithm and comprises N1 bytes, whereby all bytes of the information section were encrypted regardless of the lengths of the PES packet or the header section.

Then the encrypted PES packets can be sliced and placed in the TS packets of a standard length, if required by the application. The slicing is made so that after the header HT1 of the first TS packet there is immediately the header PH of the PES

packet, and immediately after that so many encrypted bytes of the section P'(N1) are placed in the information section T1' that the TS packet is filled. After the header HT2 of the next TS packet the process to fill the information section T2' with encrypted bytes is continued. In the last TS packet the situation is most often such that the there are less encrypted bytes of the PES packet than the length of the information section Tn' would allow. The packet is made into the standard length by increasing the length of the header HTn. This is made so that the required number of stuffing bytes are added to the adaptation field of the header. TS packets generated in this way will not be further encrypted.

The embodiment of figure 5 differs from that of figure 4 in that no encryption is made on the PES level, but block encryption is made on the TS level. The encryption is made with an accuracy of the block size, or alternatively it is made as a partly double encryption, whereby bytes of a packet outside the encryption block multiple can be encrypted. According to figure 5 the PES packet formed by the header HP and the information section P-(N1) comprising N1 bytes is sliced without encryption into n TS packets, whereby each TS packet comprises a header (HT1... Htn) and an information section (T1...Tn), in which the N1 bytes of the PES packet are placed. The TS packets are made equally long by adding stuffing bytes in the header when required. Then a block encryption is made on the TS packets, either on the multiple of the encryption block or on the whole section to be encrypted. The header blocks HT1 - Htn have varying lengths and do not belong to the region which is encrypted. But the remaining bytes belong to the region T1 - Tn to be encrypted, which after encryption will be in the form T1' - Tn', and each byte of these packets will be encrypted regardless of the length of the header section, if the encryption is partly made as double encryption, whereby also bytes outside the encryption block multiple can be encrypted. A part of the bytes will not be encrypted, if the encryption is made only with an accuracy of the block size. This is due to the fact that the information part of each TS packet is not necessarily a multiple of the encryption block. It is to be noted that the TS packet as a whole has a fixed length, but the header may have a varying length.

In the following we consider different means to make the block encryption. Figure 6 shows the functional principle of a so called chained block encryption. The section to be encrypted of the TS packet is divided in blocks B0 - Bn having a size of the encryption block, whereby the remainder R is shorter than the encryption block. The encryption device processes a whole TS packet at a time, and the encryption is started from the end, so that first the block Bn is encrypted by block encryption. The part 'Encrypt' performing the encryption generates an encrypted block Bn'. All block encryptors use a certain key to perform the block encryption, and for the sake of simplicity the key may be the same for all block encryptors. The remainder R will be moved without encryption to its own place in the encrypted TS packet. Bn' is moved through the XOR function of the block Bn-1, and thus the block encryption of block Bn-1 has as input Bn' XOR Bn-1. When we in the figure move towards the beginning of the TS packet's block to be encrypted, we will notice that the same operation is repeated for each block until we reach the beginning of the section to be encrypted. The header section HEADER is moved as such to the beginning of the encrypted TS packet.

Figure 7 shows correspondingly how the encryption of figure 6 is decrypted, whereby the first encrypted block B0' is decrypted in the decryption block 'Decrypt', and an XOR operation is performed on the result and the encrypted block B1', which at the same time produces the decrypted block B0. This result B0 is at the same time the input of the next block decryptor. Correspondingly we move block by block until we reach the end of the encrypted section of the TS packet. The remainder R was not encrypted originally, so it is already decrypted.

Figure 8 shows a so called unchained version of the block encryption in figure 6. In this version all blocks B0...Bn of the TS packet are encrypted separately, and it is not necessary to start the encryption from the end of the TS packet, but the encryption can be performed separately on each block. Starting from the beginning each block B to be encrypted is directed to the input of the encryption block 'Encrypt', and then we obtain an encrypted block B', which is placed in the encrypted TS packet at the same place as the not encrypted block B. The remainder R is not encrypted.

Figure 9 shows the decryption of the encryption made by the method according to figure 8. The decryption is made in a way which is as straightforward as the encryption. In turn each encrypted block B0'...Bn' is directed, starting from the beginning of the packet or from the block B0', to the input of the decryption block 'Decrypt', which at its output provides the original decrypted block B0...Bn, which is placed in the TS packet at the place of the corresponding encrypted block B'. The remainder R as such is placed directly to its place.

The third embodiment of the invention according to figure 10 performs the encryption both on the PES level, where byte encryption is performed, and on the TS level, where block encryption is performed with an accuracy of the block size or alternatively partly doubled encryption is per-

formed, which also incorporates the bytes outside the multiple of the encryption block. The encryption according to this embodiment is thus a combination of the encryption methods of the embodiments 1 and 2. Thus we first perform a byte encryption on the PES packet PH + P(N1), and the PES packet PH + P'(N1) containing the encrypted bytes is sliced into TS packets. In this way we obtain on the PES level encrypted TS packets, the sections of which now will be encrypted were already encrypted on the PES level. Then we perform the same operations as in figure 5 on the TS packets already encrypted on the PES level. All bytes of the TS packet's section to be encrypted are encrypted irrespective of whether the operation is according to the block size or a partly doubled encryption.

Figure 11 shows a practical way to realize the third embodiment shown in figure 10. The processing is first made on a whole PES packet of the program stream which is byte encrypted by the PRBS algorithm or a corresponding algorithm. The PRBS block means that at the input of the PRBS block there is an XOR operation on the number generated by the block itself. At the beginning of the PES packet the PRBS algorithm is initialized with an INIT initializing number, and then the PES packet is processed byte by byte, proceeding to the end of the whole PES packet. The initialization is made utilizing an encryption key in one way or another. Then the PES packet is sliced into TS packets for the transport stream, each TS packet getting its own header HT. If the combined length of the headers of the TS packet and the PES packet exceeds the last part of the TS packet, then after the first TS packet header and possibly also after the second TS packet header there is further included also PES packet header information, so that all those bytes which are outside the TS packet header do not belong to the region to be encrypted. In any case the PES packet encrypted with the PRBS algorithm is sliced into TS packets so that a plurality of TS packets are obtained from one PES packet, each TS packet having a header, several encrypted blocks B0' - Bn', and an encrypted remainder R'. These TS packets are then encrypted according to the chained encryption of figure 6, whereby the result is a plurality of encrypted blocks B0" - Bn" and an encrypted remainder R'.

The keys are transmitted from the encrypting device to the decrypting device in the digital bit stream, and the transport messages of the keys are encrypted by their own algorithms. All 'Encrypt' and 'Decrypt' blocks usually use the same key. This same key is also used in the initialization (INIT) of the PRBS block. Even if the same keys are used in both the block encryption and the byte encryption according to figures 10 and 11 when the combined block and byte encryption is performed, there are four possibilities to use the encryption keys:

1. The byte encryptor uses the same key as the block encryptor;
2. The byte encryptor and the block encryptor use different keys;
3. The blocks of the block encryptor have the same keys available; and
4. The blocks of the block encryptor can use different keys for all blocks, if this is considered necessary.

The operator can select which option to use, based on his/her own requirements.

Figure 12 shows the decryption of the encryption according to figure 11. The TS packets are first decrypted as is shown in figure 7. As a result we obtain blocks which are still encrypted by the byte encryption. Therefore each block B0' - Bn' and the remainder R' encrypted by the PRBS algorithm are further decrypted according to the PRBS algorithm so that we obtain the original TS packet. The initialization of the PRBS packet is made only at the beginning of the PES packet.

Figures 10 and 11 show how the encryption is made first on the PES packet, which then is sliced into TS packets and finally the TS packets are also encrypted by block encryption. These two encryptions can be performed in one process shown in figure 13. According to this process the PES packets are first sliced into TS packets in the way described above. Then both the byte encryption and the block encryption are performed in one process. The advantage of this arrangement is that, if required, the encryption can be made at the same time in the encryption device. The initialization of the PRBS algorithm, which utilizes an encryption key, is made at the beginning of the PES packet. The PRBS block means that in the input of the PRBS block there is an XOR operation on the number generated by the block itself. The data stream encrypted according to figure 13 can be decrypted in the way shown in figure 12.

When the encryption is made on the TS level we can select whether to encrypt all bytes outside the TS packet header, or to leave also the header of the PES packet outside the TS packet encryption. It is preferred to not encrypt the headers of the PES packet and the TS packet. Then the encrypted packet can be easily processed in the encrypted form, without having an intermediate decryption of the packets.

Figures 6 to 9 and 11 to 13 show the blocks 'Encrypt' and 'Decrypt' of the block encryption and the block 'PRBS' of the byte encryption. It must be noticed that block encryption keys and byte encryption keys are essential to these operations. All 'Encrypt' and 'Decrypt' blocks usually use the same key. This same key is also used when the

PRBS block is initialized (INIT). The keys are transmitted from the encrypting device to the decrypting device within the digital bit stream, and the transport messages of the keys are encrypted by their own algorithms.

According to the basic idea of the invention the encryption is made either on the PES packet level, on the TS packet level, or on both levels. The decryption made in the receiver is the inverse process to the encryption and according to the above a person skilled in the art can realize it in a clear-cut manner. From certain bits in the header of the TS packet the decryption device in the receiver will know on which level the decryption is made and which encryption method is used, and then it can select the correct decryption algorithm. Because the header of the PES packet follows the header of the TS packet this header can also contain the information about the encryption method if the encryption is made only on the PES level. A practical advantage of the invention is that all different encryption methods can be realized by he same device arrangements both in the receiver and in the transmitter. Although the byte encryption and the block encryption are different and require their own circuits arrangements, with suitable arrangements the circuit can be made to operate according to either encryption method or as a combination of these. When the encryption according to the third embodiment is decrypted the block decryption and byte decryption is made on the TS packets already before they are decomposed into PES packets. The encryption method according to the first embodiment of the invention can be used in such applications where only PES packets are available at the transmitting end, but the transmission should be encrypted in any case.

Claims

1. A process for encryption of a bit stream containing digital video, audio and data information, whereby the bit stream is formed by forming each information into packets comprising a first header and a first information section formed by information bytes, whereby for the transmission each packet is divided into second information sections of transport stream packets of a fixed length, whereby a packet may be divided into several information sections of the transport stream packet, **characterized** in that
   - in the encryption of the packets a first encryption key is used in a bit or byte based encryption algorithm, whereby the encryption is made with an accuracy of one bit or one byte at a time, and
   - in the encryption of transport stream packets a second encryption key is used in a block based encryption algorithm, whereby the encryption of the section to be encrypted is made by blocks which contain several bytes of a transport stream packet.

2. The process of claim 1, **characterized** in that the encryption is made only on packets or parts of the packets, but the transport stream packets are not encrypted.

3. The process of claim 1, **characterized** in that the encryption is made only on a transport stream packet or a part of it, but the packets are not encrypted.

4. The process of claim 3, **characterized** in that the transport stream packet is block encrypted and further also encrypted on a bit or byte basis.

5. The process of claim 1, **characterized** in that both packets and transport stream packets or their parts are encrypted.

6. The process of claim 1 or 5, **characterized** in that the block encryption of the transport stream packets is performed only on that section of the packet to be encrypted which has a length of the encryption block multiple.

7. The process of claim 1, 3 or 5, **characterized** in that the block based encryption algorithm is used over the whole section to be encrypted, so that first a number of bytes corresponding to a multiple of the encryption block is encrypted, and that during the second encryption phase a multiple of the encryption block is formed by performing the encryption also both on bytes of a section shorter than the block length and on a desired number of bytes of the section already once encrypted.

8. The process of claim 1 or 2, **characterized** in that the packet encryption algorithm is the PRBS algorithm (Pseudo Random Bit Sequence) known per se.

9. The process of claim 1 or 2, **characterized** in that the packet encryption algorithm does not limit the length of the section to be encrypted, but that it can be of any length with an accuracy of a byte or a bit.

10. The process of claim 2, **characterized** in that the packet header is not encrypted.

11. The process of claim 1, **characterized** in that information about whether the packets, the transport stream packets or both are encrypted is included in the transport stream packet header.

12. The process of claim 1, **characterized** in that the packet header and the transport stream header are not encrypted.

13. The process of claim 4 or 5, **characterized** in that the first and second encryption keys are the same.

14. A process for decryption of a bit stream comprising digital information, whereby the bit stream is formed by forming each information into packets comprising a first header and a first information section formed by information bytes, whereby for the transmission each packet is divided into second information sections of transport stream packets of a fixed length, whereby a packet may be divided into several information sections of the transport stream packet, **characterized** in that
    - from the transport stream packet headers it is identified whether the packets, the transport stream packets or both are encrypted,
    - in the decryption of the packets there is used a decryption algorithm on a bit or byte basis, whereby the decryption is made with an accuracy of one bit or one byte at a time, and
    - in the decryption of the transport stream packets there is used a decryption algorithm on a block basis, whereby the encrypted section is decrypted one block at a time, the block containing several bytes of a transport stream packet.

15. The process of claim 14, **characterized** in that when both the packets and the transport stream packets are encrypted then their decryption is made in one process which processes at most one block of the block encryptor at a time.

16. The process of claim 14, **characterized** in that when it is identified from the transport stream packet header that the block encryption algorithm covers the length of the second information section, then first a number of bytes corresponding to a multiple of the decryption block is decrypted once, and in a second decryption phase a multiple of the decryption block is formed so that the decryption includes both bytes of a section shorter than one block and a desired number of bytes from the region already once decrypted.

17. The process of claim 14, **characterized** in that the decryption is not made on the packet header or transport stream packet header which is not encrypted.

18. A device to encrypt a received bit stream comprising digital information comprises means to form packets of a first information section comprising a first header and information bytes of each kind of information, means to slice for transmission each packet into second information sections of transport stream packets of a fixed length, the transport stream packets comprising a second header and said second information section, whereby the packet can be divided into several information sections of the transport stream packet, **characterized** in that it further comprises:
    - means to encrypt the packets using a first encryption key with a bit or byte based encryption algorithm, whereby the encryption is made with an accuracy of one bit or byte at a time; and
    - means to encrypt the transport stream packets using a second encryption key with a block based algorithm, whereby the section to be encrypted is made on a block comprising several transport stream packets at a time.

19. A device to decrypt a received bit stream comprising digital information, the stream being provided by forming packets of each information, the packets comprising a first information section comprising a first header and information bytes, by dividing each packet for transmission into information sections of transport stream packets of fixed length, the transport stream packets comprising a second header and said second information section, whereby the packet can be divided into several information sections of the transport stream packet, **characterized** in that it comprises
    - means that identifies from the transport stream packet header information whether the packets, transport stream packets of both are encrypted, and that further identifies the used encryption keys;
    - means to decrypt the packets with a bit or byte based decryption algorithm, whereby the decryption is made with an accuracy of a bit or a byte at a time,
    - means to decrypt the packets with a decryption algorithm based on blocks, whereby the encrypted section is de-

crypted one block a time containing several transport stream bytes.

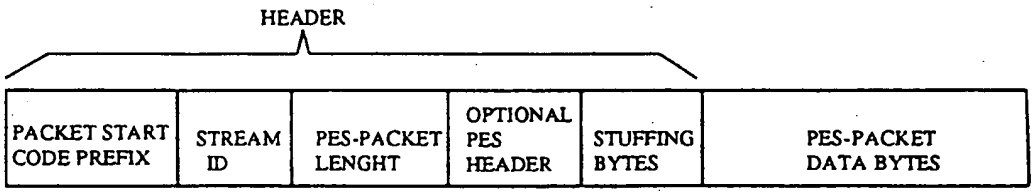

FIG. 1

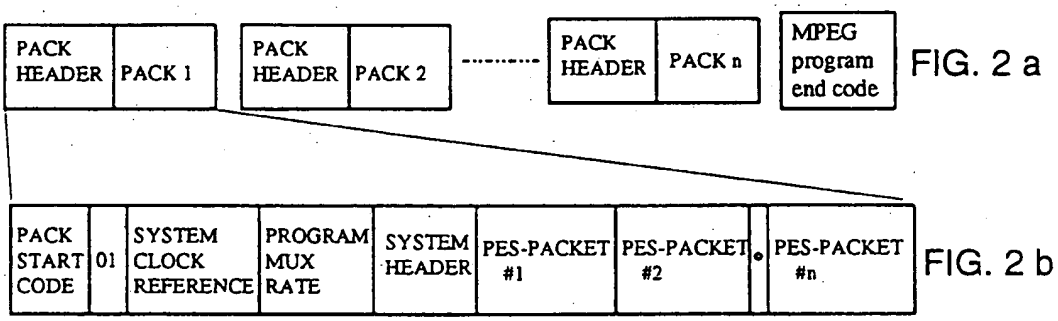

FIG. 2 a

FIG. 2 b

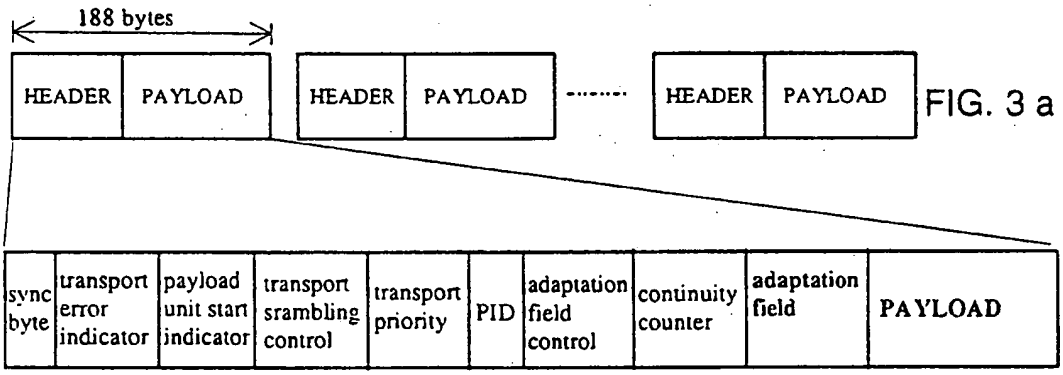

FIG. 3 a

FIG. 3 b

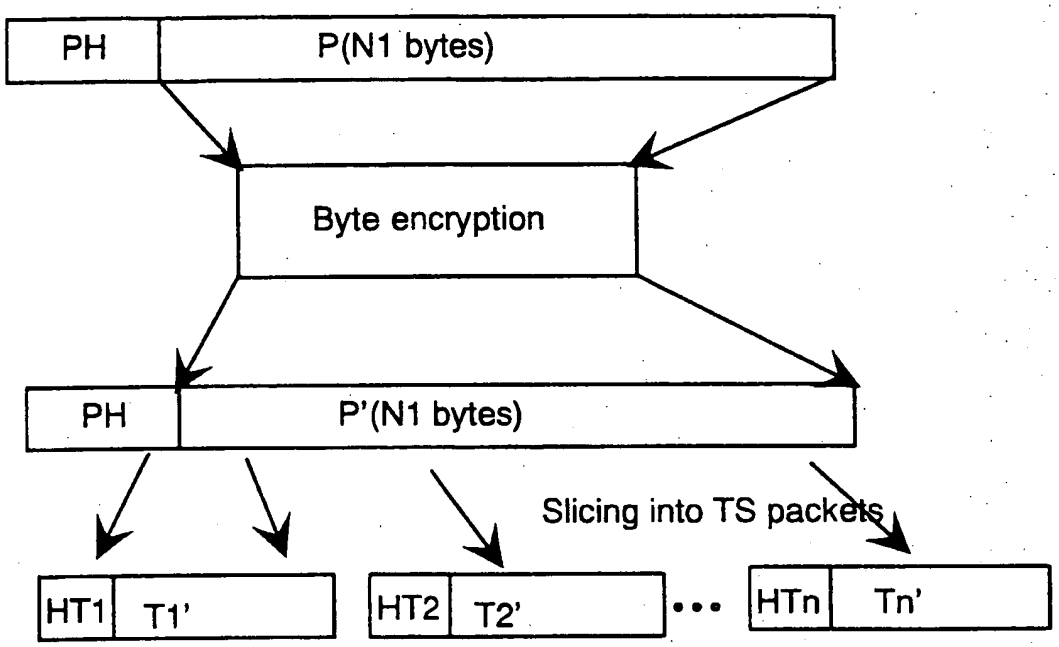
PH
P(N1 bytes)
Byte encryption
PH
P'(N1 bytes)
Slicing into TS packets
HT1
T1'
HT2
T2'
HTn
Tn'

FIG. 4

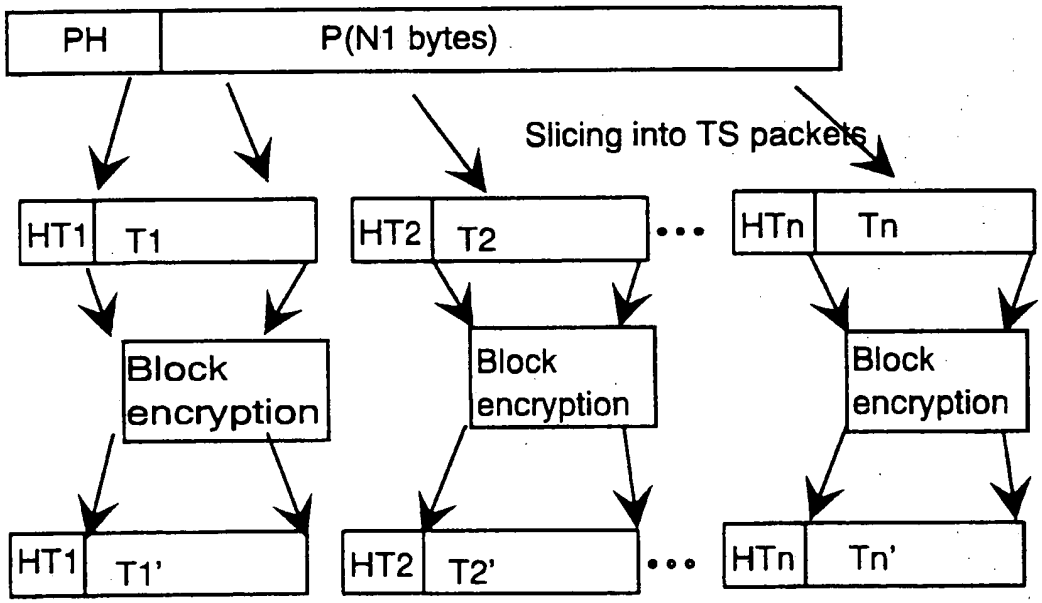
PH
P(N1 bytes)
Slicing into TS packets
HT1
T1
HT2
T2
HTn
Tn
Block encryption
Block encryption
Block encryption
HT1
T1'
HT2
T2'
HTn
Tn'

FIG. 5

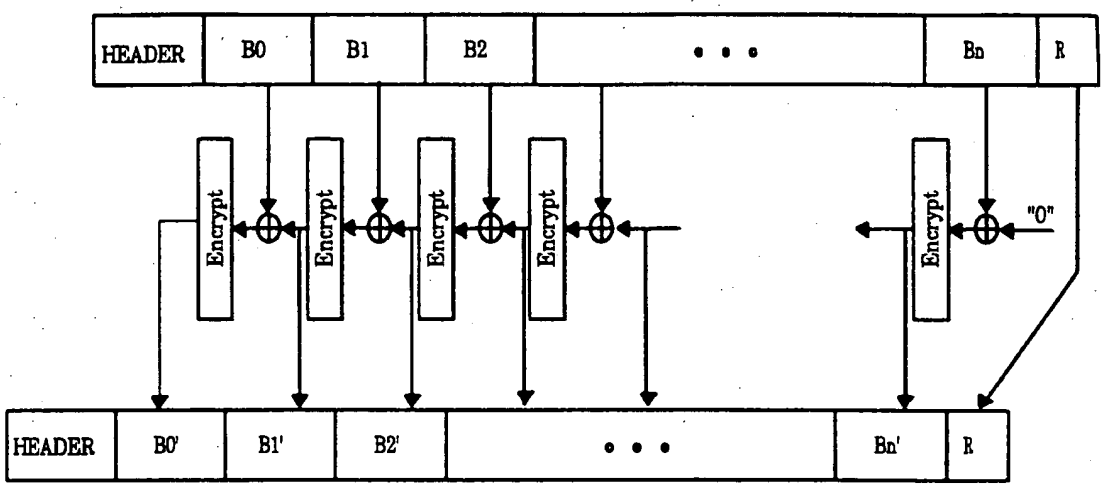
HEADER
B0
B1
B2
• • •
Bn
R
Encrypt
Encrypt
Encrypt
Encrypt
Encrypt
"0"
HEADER
B0'
B1'
B2'
• • •
Bn'
R

FIG. 6

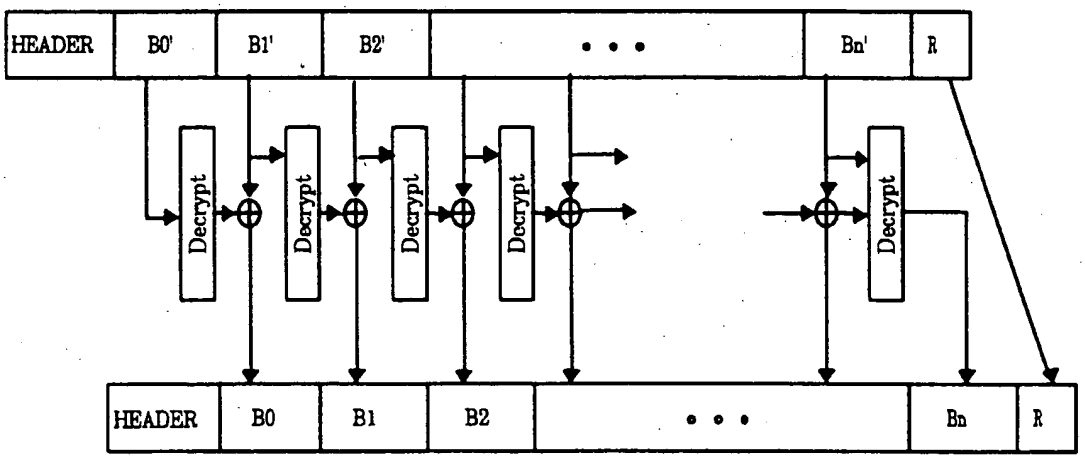
HEADER
B0'
B1'
B2'
• • •
Bn'
R
Decrypt
Decrypt
Decrypt
Decrypt
Decrypt
HEADER
B0
B1
B2
• • •
Bn
R

FIG. 7

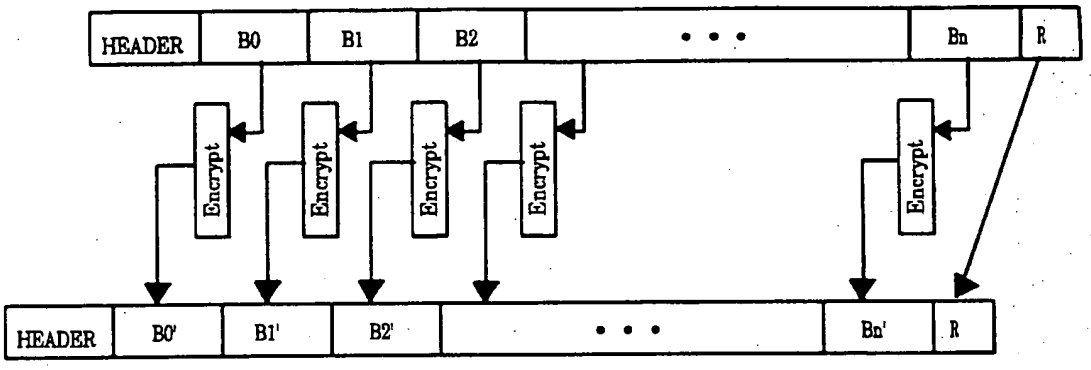
HEADER
B0
B1
B2
• • •
Bn
R
Encrypt
Encrypt
Encrypt
Encrypt
Encrypt
HEADER
B0'
B1'
B2'
• • •
Bn'
R

FIG. 8

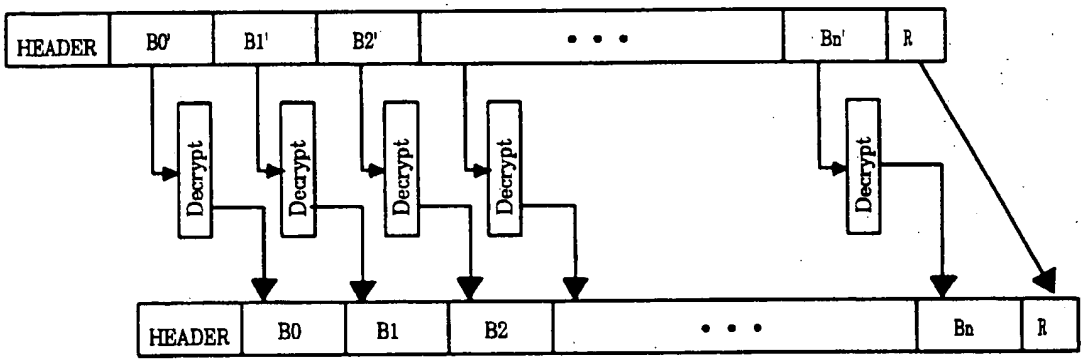
HEADER
B0'
B1'
B2'
• • •
Bn'
R
Decrypt
Decrypt
Decrypt
Decrypt
Decrypt
HEADER
B0
B1
B2
• • •
Bn
R

FIG. 9

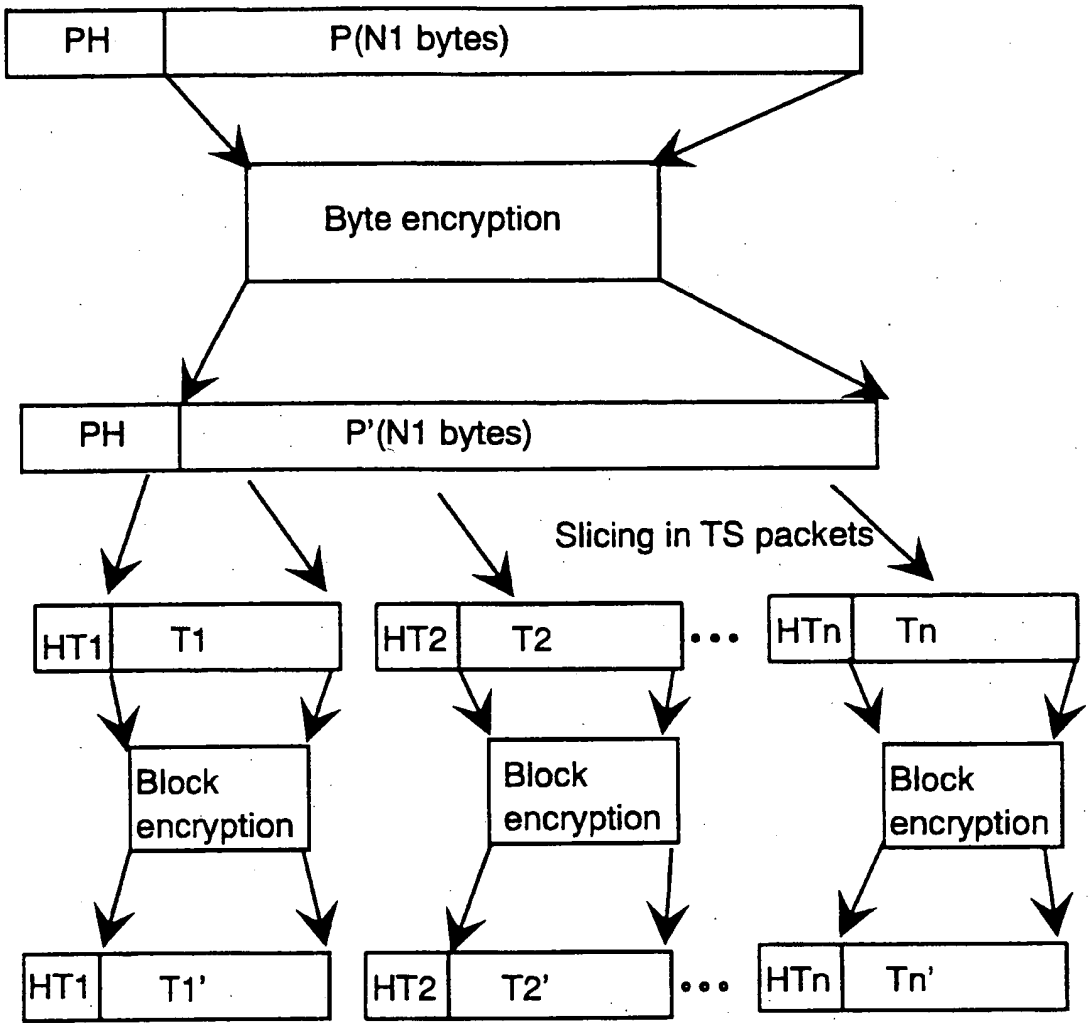
PH
P(N1 bytes)
Byte encryption
PH
P'(N1 bytes)
Slicing in TS packets
HT1
T1
HT2
T2
HTn
Tn
Block encryption
Block encryption
Block encryption
HT1
T1'
HT2
T2'
HTn
Tn'

FIG. 10

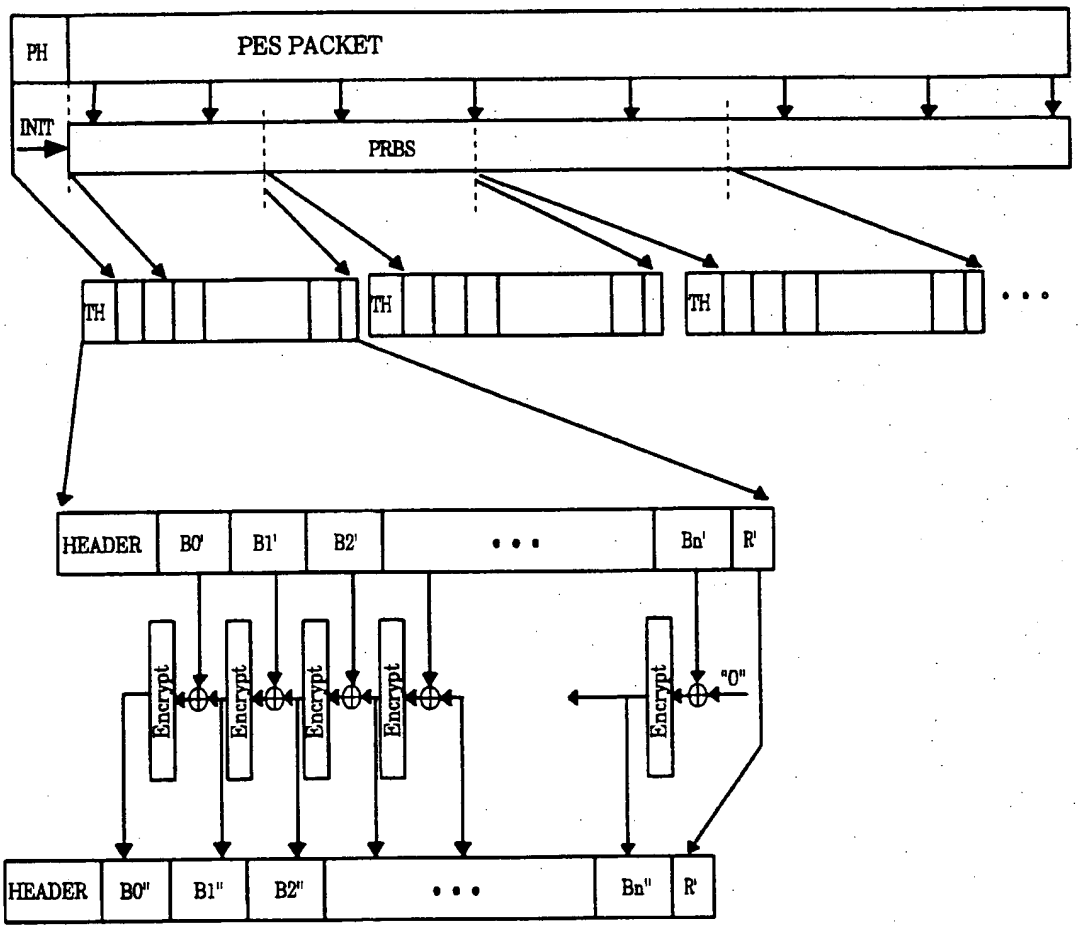
PH
PES PACKET
INIT
PRBS
TH
TH
TH
HEADER
B0'
B1'
B2'
Bn'
R'
Encrypt
Encrypt
Encrypt
Encrypt
Encrypt
"0"
HEADER
B0"
B1"
B2"
Bn"
R'

FIG. 11

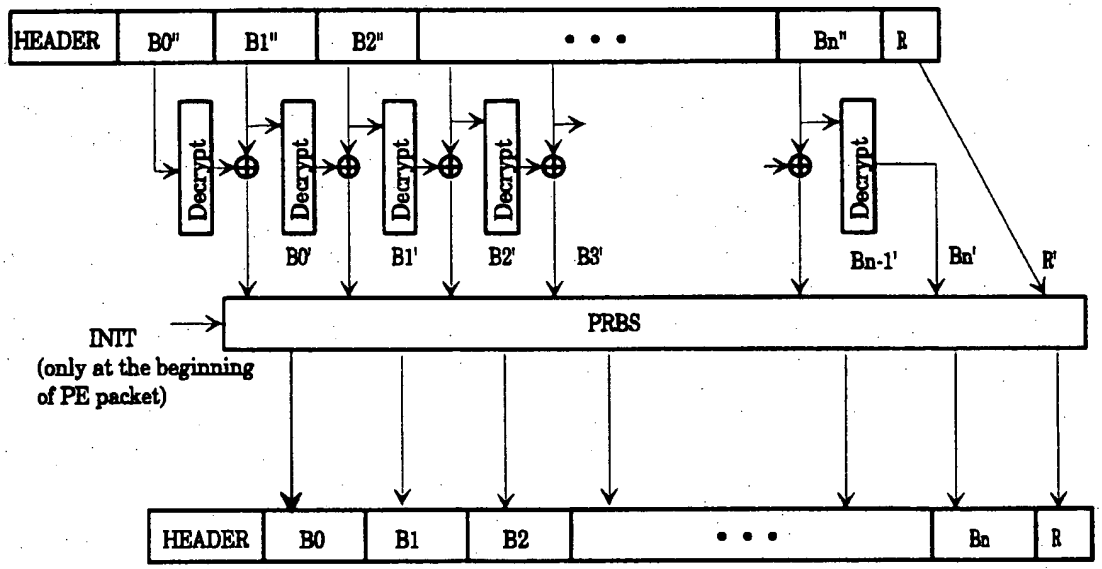
HEADER
B0"
B1"
B2"
• • •
Bn"
R
Decrypt
Decrypt
Decrypt
Decrypt
Decrypt
B0'
B1'
B2'
B3'
Bn-1'
Bn'
R'
PRBS
INIT
(only at the beginning
of PE packet)
HEADER
B0
B1
B2
• • •
Bn
R

FIG. 12

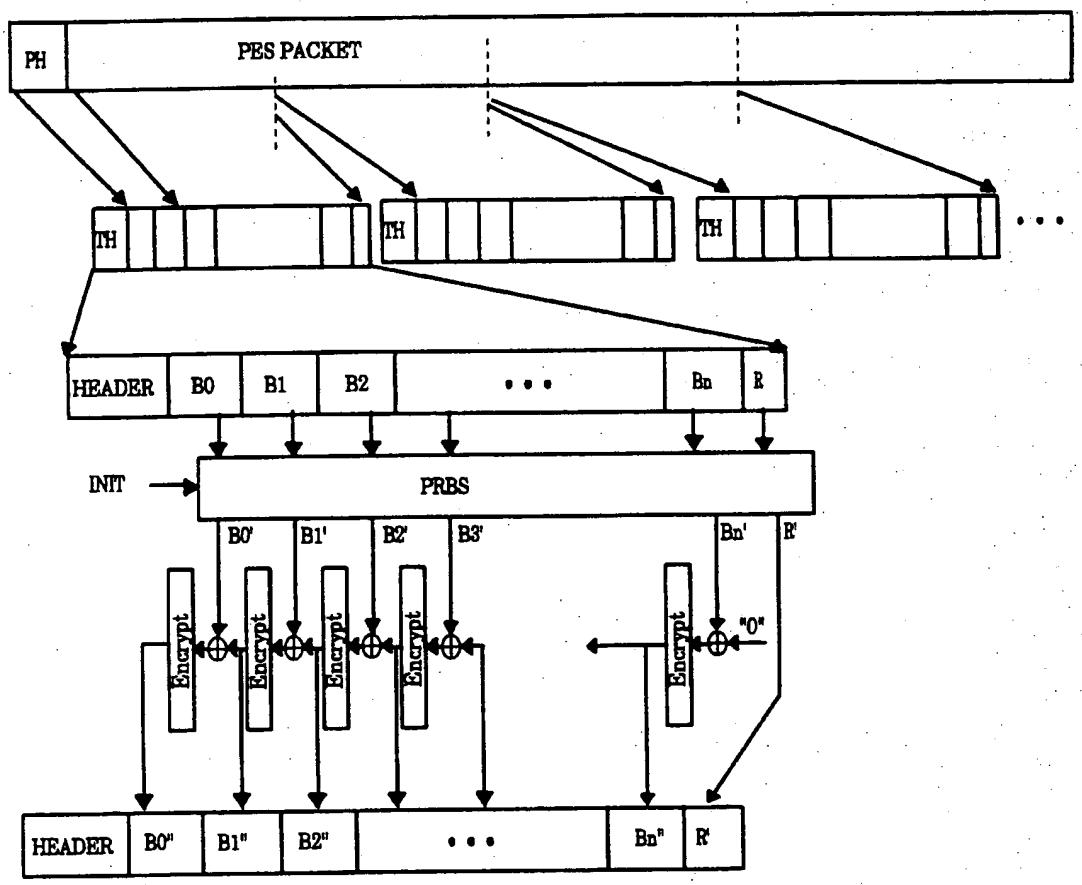
PH
PES PACKET
TH
TH
TH
HEADER
B0
B1
B2
Bn
R
INIT
PRBS
B0'
B1'
B2'
B3'
Bn'
R'
Encrypt
Encrypt
Encrypt
Encrypt
Encrypt
"0"
HEADER
B0"
B1"
B2"
Bn"
R'

FIG. 13